

# Non-linear Adaptive Controllers with Linear Dynamic Compensator and Neural Network

LUNGU MIHAI

Avionics Department

University of Craiova, Faculty of Electrotechnics

Blv. Decebal, No.107, Craiova, Dolj

[lma1312@yahoo.com](mailto:lma1312@yahoo.com)

ROMANIA

LUNGU ROMULUS

Avionics Department

University of Craiova, Faculty of Electrotechnics

Blv. Decebal, No.107, Craiova, Dolj

[romulus\\_lungu@yahoo.com](mailto:romulus_lungu@yahoo.com)

ROMANIA

*Abstract:* - The paper presents complex adaptive non-linear systems with one input and one output which are based on dynamic inversion. Linear dynamic compensator makes the stabilization command of the linearised system using as input the difference between closed loop system's output and the reference model's output (command filter). The state vector of the linear dynamic compensator, the output and other state variables of the control system are used for adaptive control law's obtaining; this law is modeled by a neural network. The aim of the adaptive command is to compensate the dynamic inversion error. Thus, the command law has two components: the command given by the linear dynamic compensator and the adaptive command given by the neural network. For estimation the dynamic compensator's state, the non-linear adaptive controller may have a linear reduced order observer. As control system one chooses the non-linear model of helicopter's dynamics in longitudinal plain. The reference model is linear. One obtains the structure of the adaptive control system of the pitch angle and Matlab/Simulink models of the adaptive command system's subsystems. Using these, some characteristics families are obtained. These describe the adaptive command system's dynamics with linear or non-linear actuator.

*Key-Words:* compensator, neural network, dynamic inversion.

## 1 Introduction

The complexity and incertitude that appear in the non-linear and instable phenomena are the main reasons that require the projecting of non-linear adaptive structures for control and stabilization; in these cases the linear models are far from a good describe of the flying object's dynamic. Another reason is the non-linear character of the actuators (because of the saturation and/or their displacement velocity). The observers must be easily adaptable and their project algorithms must allow the estimation of the state of the flying object even in the case of their failure or no use of the damaged sensors' signals. In these situations it's good to use the real time adaptive control based on neural networks and dynamic inversion of the unknown or partial known nonlinearities from the dynamic model of the flying object [1]. The train of the neural networks is based on the signals from state

observers; these observers get information about the control system's error [2], [3], [4].

## 2 Dynamic SISO systems

Let's consider the dynamic system (A) with single input and single output (SISO) described by equations

$$\dot{x} = f(x, u), y = h(x), \quad (1)$$

with  $x(n \times 1)$ ,  $n$  - unknown  $f$  and  $h$  - unknown nonlinear functions,  $u$  and  $y$  - measurable.

One projects an adaptive control law  $v$  after (in rapport with) the output; the neural network (NN) models a function that depends on the values of input and output of A at different time moments so that  $y(t)$  follows the finite  $\bar{y}(t)$ . The feedback linearization may be made through transformation [5]

$$v = \hat{h}_r(y, u), \quad (2)$$

with  $v$  is the pseudo-command signal and  $\hat{h}_r(y, u)$  – the best approximation of  $h_r(x, u) = h_r(x(y), u)$ .

Equation (2) is equivalent with the following one

$$u = \hat{h}_r^{-1}(y, v). \quad (3)$$

If  $\hat{h}_r \equiv h_r$ , one yields  $y^{(r)} = v$ ; otherwise  $(\hat{h}_r \neq h_r)$

$$y^{(r)} = v + \varepsilon, \quad (4)$$

where

$$\varepsilon = \varepsilon(x, u) = h_r(x, u) - \hat{h}_r(y, u) \quad (5)$$

is the approximation of function  $h_r$  (inversion error). Assessing  $y$  to follow  $\bar{y}$ , then  $v$  has the form [5], [6], [7]

$$v = \bar{y}^{(r)} + v_{pd} - v_a + \bar{v}, \quad (6)$$

where  $v_{pd}$  is the output of the dynamic linear compensator for stabilization, used for linearised dynamic (4), with  $\varepsilon = 0, v_a$  – the adaptive command that must compensate  $\varepsilon$  and  $\bar{v}$  has the form [8]

$$\bar{v} = k_z \left( \|\hat{Z}\|_F + \bar{Z} \right) \frac{\bar{E}}{\|\bar{E}\|} \|\hat{E}\| + k_v \bar{E}, \quad (7)$$

with  $k_z, k_v > 0$  gain constants,  $\|\hat{Z}\|_F$  – the Frobenius norm of matrix  $\hat{Z}, \bar{Z}$  – the ideal matrix of the neural network and  $\bar{E} = \hat{E}P\bar{B}$ , with  $\hat{E}, P$  and  $\bar{B}$  – matrices. The derivative  $\bar{y}^{(r)}$  is introduced for the conditioning of the dynamic error  $\tilde{y} = \bar{y} - y$ . This derivative is given by a reference model (command filter) [5].  $\bar{y}^{(r)}$  may be cumulated with other signals and it results the component  $v_r$  of form (11).

Considering

$$Y^T = [y \quad \dot{y} \quad \dots \quad y^{(r-1)}], Z^T = [v \quad \dot{v} \quad \dots \quad v^{(p)}], \quad (8)$$

$$\lambda^T = [\lambda_0 \quad \lambda_1 \quad \dots \quad \lambda_{r-1}], b^T = [b_0 \quad b_1 \quad \dots \quad b_p],$$

with  $b_i, i = \overline{0, p}, \lambda_j, j = \overline{0, r-1}$  the coefficients of the numerator and denominator of the transfer function for the system with input  $u_n$  and output  $y$ , the linear system with input  $v$  and output  $y$  is described by equation

$$y^{(r)} = -\lambda^T Y + b^T Z + \varepsilon. \quad (9)$$

If  $p = 0$ , then  $Z = v, b = b_0$  and the previous equation becomes

$$y^{(r)} = -\lambda^T Y + b_0 v + \varepsilon. \quad (10)$$

In the particular case  $y^{(r)} = \bar{y}^{(r)}$ , one obtains

$$v_r = \frac{1}{b_0} (\bar{y}^{(r)} + \lambda^T \bar{Y}). \quad (11)$$

The compensator may be described by state equations

$$\dot{\zeta} = A_c \zeta + b_c e, \quad (12)$$

$$v_{pd} = c_c \zeta + d_c e,$$

where  $\zeta$  has at least dimension  $(r-1)$ ,

$$e = \tilde{y} = ce, e^T = [e \quad \dot{e} \quad \dots \quad e^{(r-1)}], \quad (13)$$

$$c = [1 \quad 0 \quad 0 \quad \dots \quad 0]_{1 \times r}.$$

The state equation of the linear subsystem with input  $(v + \varepsilon)$  and output  $y$  is

$$\dot{x} = Ax + b(v + \varepsilon), v = v_{pd} - v_a + \bar{v}, \quad (14)$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix}_{r \times r}, b = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}_{r \times 1}. \quad (15)$$

The stable state  $\bar{x} (\bar{x} = v = \varepsilon = 0)$  verifies equation

$A\bar{x} = 0$  and, taking into account (14), leads to the equation of the error vector  $e \equiv \tilde{x} = \bar{x} - x$ ,

$$\dot{e} = Ae - bv_{pd} + b(v_a - \bar{v} - \varepsilon). \quad (16)$$

With notations

$$E = \begin{bmatrix} e \\ \zeta \end{bmatrix}, \bar{A} = \begin{bmatrix} A - d_c bc & -bc_c \\ b_c c & A_c \end{bmatrix}, \bar{b} = \begin{bmatrix} b \\ 0 \end{bmatrix}, \bar{C} = \begin{bmatrix} c & 0 \\ 0 & I \end{bmatrix}, \quad (17)$$

where  $I$  is the identity matrix with the  $\zeta$ 's dimension, one obtains

$$\dot{E} = \bar{A}E + \bar{b}(v_a - \bar{v} - \varepsilon), z = \bar{C}E; \quad (18)$$

$A_c, b_c, c_c, d_c$  from (12) are calculated such that  $\bar{A}$  be a Hurwitz matrix.

For the estimation of the error vector  $E$  one may use a linear state observer of order  $(2r-1)$  described by equations

$$\dot{\hat{E}} = \bar{A}\hat{E} + L(z - \hat{z}), \hat{z} = \bar{C}\hat{E}, \quad (19)$$

with the gain matrix  $L$  calculated so that matrix  $\tilde{A} = (\bar{A} - L\bar{C})$  is stable. Considering  $w$  – the sensor measure error,  $y_m$  – the measured value of  $y$ , then  $\tilde{y}_m = \bar{y} - y_m = \tilde{y} + w$  and the compensator's equations become

$$\dot{E} = \bar{A}E + \bar{b}(v_a - \bar{v} - \varepsilon) + Gw, z = \bar{C}E + Hw. \quad (20)$$

with  $H^T = [1 \quad 0], G^T = [-bd_c \quad b_c]$ .

If state  $\zeta$  of the compensator is known, one uses a reduced order observer for estimation of vector  $e$

$$\dot{\hat{e}} = \bar{A}\hat{e} + L_r(z_1 - \hat{z}_1), z_1 = c\hat{e}. \quad (21)$$

The gain matrix  $L_r$  is obtained so that matrix  $\tilde{A} = (\bar{A} - L_r c)$  is stable. With vectors  $\hat{e}$  and  $\zeta$  vector  $\hat{E}^T = [\hat{e} \quad \zeta]$  is obtained.

The signal  $\bar{E} = \hat{E}^T P \bar{b}$  is used for neural network's adapting; the weights  $\hat{W}$  and  $\hat{V}$  are obtained with equation

$$\begin{aligned} \dot{\hat{W}} &= -\Gamma_w [2(\hat{\sigma} - \hat{\sigma}' \hat{v}^T \eta) \hat{E}^T P \bar{B} + k(\hat{W} - \hat{W}_0)], \\ \dot{\hat{V}} &= -\Gamma_v [2\eta \hat{E}^T P \bar{B} \hat{\sigma}' + k(\hat{V} - \hat{V}_0)], \end{aligned} \quad (22)$$

where the role of  $\bar{B}$  is played by  $\bar{b}$ . In (22)  $\sigma$  is the sigmoid function

$$\sigma(z) = \frac{1}{1 + e^{-az}}, \quad (23)$$

$\hat{\sigma}' = \left. \frac{d\hat{\sigma}(z)}{dz} \right|_{z=z_0}$  is the Jacobian of vector  $\hat{\sigma}, \hat{W}_0$  and  $\hat{V}_0$

the initial values of weights  $\hat{W}, \hat{V}, \Gamma_w, \Gamma_v > 0, k > 2(k_1^2 + \gamma_1^2 \|P\bar{B}\|^2), k_1 = k_2 \alpha_1 + \|P\bar{B}\| \gamma_1, k_2 = \|P\bar{B}\| + \|\bar{P}\bar{B}\|,$

$P$  and  $\bar{P}$  – the solutions of Liapunov equations  $\bar{A}^T P + P \bar{A} = -Q, \tilde{A}^T \tilde{P} + \tilde{P} \tilde{A} = -\tilde{Q}.$  (24)

$P$  from the signal used for the neural network's adapting is the solution of first equation (24) with  $\bar{A} = (A - d_c bc).$

Second output of the compensator ( $\tilde{y}_a$ ) is used for obtaining of an error signal that is useful for adapting of the neural network's weights.

From (4) and (6) one yields

$$y^{(r)} = \bar{y}^{(r)} + v_{pd} - v_a + \bar{v} + \varepsilon, \quad (25)$$

equivalent with the dynamic error's equation

$$\tilde{y}^{(r)} = -v_{pd} + v_a - \bar{v} - \varepsilon. \quad (26)$$

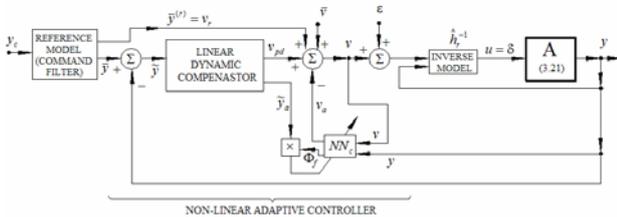


Fig.1. Automat control system with non-linear adaptive controller

According to (26) and fig.1, it results the block diagram for the dynamic error's modeling (fig.2).

Error  $\varepsilon$  may be approximated with the output of a linear neural network NN [5]

$$\varepsilon = W^T \Phi(\eta) + \mu(\eta), \|\mu\| < \mu^*, \quad (27)$$

where  $W$  is the weights' matrix for the connections between layer 2 and layer 3 (NN has 2 layers),  $\mu(\eta)$  – the reconstruction error of the function and  $\eta$  – the input vector of NN

$$\eta = [1 \quad \bar{v}_d^T(t) \quad \bar{y}_d^T(t)]^T, \quad (28)$$

where

$$\begin{aligned} \bar{v}_d^T(t) &= [v(t) \quad v(t-d) \quad \dots \quad v(t-(n_1-r-1)d)]^T, \\ \bar{y}_d^T(t) &= [y(t) \quad y(t-d) \quad \dots \quad y(t-(n_1-1)d)]^T, \end{aligned} \quad (29)$$

with  $n_1 \geq n$  and  $d > 0; v_a$  is projected so that

$$v_a = \hat{W}^T \Phi(\eta), \quad (30)$$

where  $\hat{W}$  is the estimation of  $W$ .

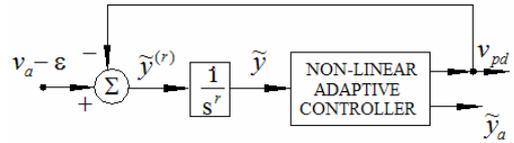


Fig.2. Block diagram of the dynamic error's modeling

### 3 Adaptive system for the helicopter pitch angle command

Lets' consider the case of nonlinear dynamic of an experimental helicopter R – 50 with one input and one output; its dynamic is (1) with

$$x_e^T = [V_x \quad \omega_y \quad \theta \quad \beta \quad V_z]^T, u = \delta, y = \theta, \quad (31)$$

where  $V_x, V_z$  are the advance velocity, respectively the vertical velocity,  $\theta$  and  $\omega_y$  – the pitch angle and the pitch angular velocity,  $\beta$  – the longitudinal control angle of the main rotor,  $\delta$  – the cyclic longitudinal input. Choosing the linearised model of helicopter [9] and, annexing the actuator's equation

$$\tau \dot{\delta} + \delta = \delta_c, \quad (32)$$

the new state vector  $x_e^T = [V_x \quad \omega_y \quad \theta \quad \beta \quad V_z \quad \delta]^T,$  input  $u = \delta_c,$  output  $y = \theta$  and state equation are obtained

$$\begin{bmatrix} \dot{V}_x \\ \dot{\omega}_y \\ \dot{\theta} \\ \dot{\beta} \\ \dot{V}_z \\ \dot{\delta} \end{bmatrix} = \begin{bmatrix} X_u & X_q & X_\theta & X_\beta & X_w & X_\delta \\ M_u & M_q & 0 & M_\beta & M_w & M_\delta \\ 0 & 0.9991 & 0 & 0 & 0 & 0 \\ B_u & -1 & 0 & B_\beta & 0 & B_\delta \\ Z_u & Z_q & Z_\theta & Z_\beta & Z_w & Z_\delta \\ 0 & 0 & 0 & 0 & 0 & -1/\tau \end{bmatrix} \begin{bmatrix} V_x \\ \omega_y \\ \theta \\ \beta \\ V_z \\ \delta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1/\tau \end{bmatrix} \delta_c. \quad (33)$$

From (33) one yields

$$\dot{y} = \dot{\theta} = \omega_y, \ddot{y} = \ddot{\theta} = M_u V_x + M_q \dot{\theta} + M_\beta \beta + M_w V_z + M_\delta \delta. \quad (34)$$

In order to relieve the non-linear function  $h_r(x, u) = h_r(x, \delta_c),$  one must express an equation that contains  $\delta_c.$  By derivation of the second equation (34) and substituting  $\dot{\omega}_y = \ddot{\theta} = \ddot{y}$  and  $\delta$  from (32) it results

$$\ddot{y} = \ddot{\theta} = M_u \dot{V}_x + M_q \ddot{\theta} + M_\beta \dot{\beta} + M_w \dot{V}_z - \frac{M_\delta}{\tau} \dot{\delta} + \frac{M_\delta}{\tau} \delta_c. \quad (35)$$

So, this equation may be completed with a non-linear component so that (35) has the form

$$y^{(r)} = h_r(r, u), h_r = \frac{d^r h}{dt^r}; \quad (36)$$

It results  $r=3.$  The control system has the form from fig.1. One chooses the reference model

$$\begin{aligned} \bar{y} &= \frac{p \omega_{r0}^2}{(s+p)(s^2 + 2\xi_0 \omega_{r0} s + \omega_{r0}^2)} y_c, \\ p &= 25, \xi_0 = 0.7, \omega_{r0} = 10 \text{ rad/s}. \end{aligned} \quad (37)$$

The transfer function for the system with input  $u_n$  and output  $y$  has the form

$$H_d(s) = \frac{b_0}{s^3 + \lambda_2 s^2} = \frac{b_0}{s^2(s + b_0)}; \quad (38)$$

One has chosen  $\lambda_2 = b_0$ . According to (8) it results  $Z = v, \lambda_0 = \lambda_1 = 0$  and equation (9) becomes

$$\ddot{y} = -b_0 \dot{y} + b_0 v + \varepsilon. \quad (39)$$

By elimination of  $\ddot{y} = \ddot{\theta}$  between the second equation (34) and (35) one yields

$$\ddot{y} = \ddot{\theta} = M_u \dot{V}_x + M_\beta \dot{\beta} + M_w \dot{V}_z - \frac{M_\delta}{\tau} \delta + \frac{M_\delta}{\tau} \delta_c + M_q (M_u V_x + M_\beta \beta + M_w V_z + M_\delta \delta) + M_q^2 \dot{\theta}. \quad (40)$$

Replacing  $\dot{y}$  given by the second equation (34) and  $\ddot{y}$  given by (40) in (39) one may identify  $v$  and  $\varepsilon$  as follows: 1)  $v$  must have the form  $v = h_r(x, u) = h_r(x, \delta_c)$ , where  $x^T = [\theta \ \dot{\theta} \ \ddot{\theta}]$ ;  $h_r$  depends on some of state vector's components; 2)  $\varepsilon$  depends on the other terms that contain  $V_x, V_z, \beta, \delta, \dot{V}_x, \dot{V}_z, \dot{\beta}$ . One obtains

$$\begin{aligned} \delta_c = u &= \frac{\tau}{M_\delta} [b_0 v - \hat{M}_q (\hat{M}_q + b_0) \dot{\theta}] = \hat{h}_r^{-1}(x, v), \\ \varepsilon &= (M_q + 2b_0) (M_u V_x + M_\beta \beta + M_w V_z + M_\delta \delta) + \\ &+ M_u \dot{V}_x + M_\beta \dot{\beta} + M_w \dot{V}_z - \frac{M_\delta}{\tau} \delta. \end{aligned} \quad (41)$$

Because  $\lambda^T = [0 \ 0 \ b_0]$  and  $\bar{Y}^T = [\tilde{y} \ \dot{\tilde{y}} \ \ddot{\tilde{y}}]$ , with (11) one obtains

$$v_r = \frac{1}{b_0} (\ddot{\tilde{y}} + b_0 \dot{\tilde{y}}). \quad (42)$$

By choosing for the compensator a proportional-derivative control law  $v_{pd} = k_p \tilde{y} + k_d \dot{\tilde{y}}$ , the law (6), in which the role of  $\bar{y}^{(r)}$  is played by  $v_r$ , becomes

$$v = k_p \tilde{y} + k_d \dot{\tilde{y}} + \frac{1}{b_0} \ddot{\tilde{y}} + \dot{\tilde{y}} - v_a + \bar{v}; \quad (43)$$

Taking into account (4) and (43) it results

$$\ddot{\tilde{y}} = -b_0 k_p \tilde{y} - b_0 k_d \dot{\tilde{y}} - b_0 \ddot{\tilde{y}} + (b_0 v_a - b_0 \bar{v} - \varepsilon). \quad (44)$$

The system that describes the dynamic of the error  $e^T = [\tilde{y} \ \dot{\tilde{y}} \ \ddot{\tilde{y}}]$  is

$$\begin{bmatrix} \dot{\tilde{y}} \\ \dot{\dot{\tilde{y}}} \\ \dot{\ddot{\tilde{y}}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -b_0 k_p & -b_0 k_d & -b_0 \end{bmatrix} \begin{bmatrix} \tilde{y} \\ \dot{\tilde{y}} \\ \ddot{\tilde{y}} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} (b_0 v_a - b_0 \bar{v} - \varepsilon). \quad (45)$$

For the calculus of coefficients  $b_0, k_p, k_d$ , one sets values for the roots of the characteristic equation

$$s^3 + b_0 s^2 + b_0 k_d s + b_0 k_p = 0; \quad (46)$$

that means to set the coefficients of this equation to be Višnegradski type;  $b_0 = 1$ .

From the expression of the control law and (12) it results that the linear dynamic compensator has order 1 with only a state variable  $S = \tilde{y} = e$ . As a consequence, conform to equation (17), one

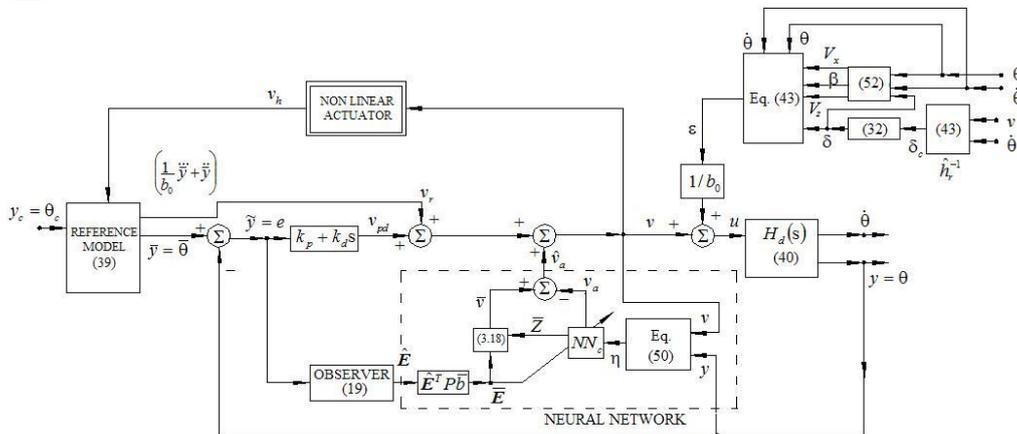


Fig.3. Block diagram of the system for automat command of the pitch angle

yields  $E^T = e^T = [\tilde{y} \ \dot{\tilde{y}} \ \ddot{\tilde{y}}]$ ,  $\bar{A} = (A - d_c bc)$  and (45) has the form (18) and  $z = \bar{C} E = ce = \tilde{y}$ . The state estimator is described by equations (19);  $\hat{E} = \hat{e}, \hat{z} = \hat{e}$ . Gain  $L$  is calculated so that matrix  $\tilde{A} = (A - LC)$  is stable.

The component  $v_a$  is calculated using the equation

$$v_a = \hat{W}^T \sigma(\hat{V}^T \eta), \quad (47)$$

with  $\hat{W}$  and  $\hat{V}$  having the form (22) and the input

vector  $\eta$  of form (28) with components (29);

$$n_1 = n = 6, r = 3, d = 0.05; \Gamma_w = 23, \Gamma_v = 12.5, k = 0.115;$$

matrices  $P, \tilde{P}$  are the solutions of equations (24);

$$\eta^T = [1 \ v(t) \ v(t-d) \ v(t-2d) \ y(t) \ y(t-d) \ y(t-2d)]. \quad (48)$$

Function  $\sigma$  is obtained using (23) and  $z = \hat{V}^T \eta$ .

The component  $\bar{v}$  is obtained using (7) and  $k_z = 0.8, k_v = 0.7, \bar{Z} = 50$ . For the calculus of  $\varepsilon$  with (41) the other state variables ( $V_x, V_y, V_z, \delta$ ) and the

derivates  $(\dot{V}_x, \dot{V}_z, \dot{\beta})$  must be known. The first, the forth and the fifth equations (33) lead to

$$\begin{aligned} \dot{V}_x &= X_u V_x + X_q \dot{\theta} + X_\theta \theta + X_\beta \beta + X_w V_z + X_\delta \delta, \\ \dot{\beta} &= B_u V_x - \dot{\theta} + B_\beta \beta + B_\delta \delta, \\ \dot{V}_z &= Z_u V_x + Z_q \dot{\theta} + Z_\theta \theta + Z_\beta \beta + Z_w V_z + Z_\delta \delta. \end{aligned} \quad (49)$$

The values of the coefficients from (33) are

$$\begin{aligned} X_u &= -0.0553, X_q = 1.413, X_\theta = -32.1731, X_\beta = -19.9033, \\ X_w &= 0.0039, M_u = 0.2373, M_q = -6.9424, M_\beta = 68.2896, \\ M_w &= 0.002, B_u = 0.0101, B_\beta = -2.1633, Z_u = -0.0027, Z \\ q &= -0.0236, Z_\theta = -0.2358, Z_\beta = -0.1233, Z_w = -0.5727, \\ X_\delta &= 11.2579, M_\delta = -38.6267, B_\delta = -4.2184, Z_\delta = 0.0698, \\ \dot{M}_\delta &= 0.5M_\delta, \dot{M}_q = 2M_q, \tau = 0.05. \end{aligned}$$

For the calculus of  $\delta$  equation (32) is used, with  $\delta_c$  of form (41). The previous equations system (49) is equivalent with the following one

$$\begin{bmatrix} \dot{V}_x \\ \dot{\beta} \\ \dot{V}_z \end{bmatrix} = \begin{bmatrix} X_u & X_\beta & X_w & X_q \\ B_u & B_\beta & 0 & B_\delta \\ Z_u & Z_\beta & Z_w & Z_\delta \end{bmatrix} \begin{bmatrix} V_x \\ \beta \\ V_z \\ \delta \end{bmatrix} + \begin{bmatrix} X_\theta & X_q & X_\delta \\ 0 & -1 & B_\delta \\ Z_\theta & Z_q & Z_\delta \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \delta \end{bmatrix}. \quad (50)$$

With this one (41) becomes

$$\begin{aligned} \varepsilon &= (M_q + 2b_0) \begin{bmatrix} M_u & M_\beta & M_w & M_\delta \end{bmatrix} \begin{bmatrix} V_x \\ \beta \\ V_z \\ \delta \end{bmatrix} + \begin{bmatrix} M_u & M_\beta & M_w \end{bmatrix} \\ &\cdot \left\{ \begin{bmatrix} X_u & X_\beta & X_w & X_q \\ B_u & B_\beta & 0 & B_\delta \\ Z_u & Z_\beta & Z_w & Z_\delta \end{bmatrix} \begin{bmatrix} V_x \\ \beta \\ V_z \\ \delta \end{bmatrix} + \begin{bmatrix} X_\theta & X_q \\ 0 & -1 \\ Z_\theta & Z_q \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} \right\} - \frac{M_\delta}{\tau} \delta. \end{aligned} \quad (51)$$

The block diagram of the reference model is the one presented in fig.4 and the block diagram of the system for automat command of the pitch angle is presented in fig.3.

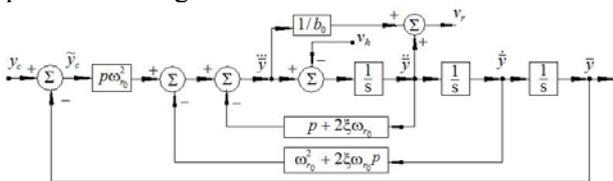


Fig.4. The block diagram of the reference model

Actuators' characteristics (time delays, nonlinearities with saturation zone) lead to adapting difficulties of the neural network. This is why a block "PCH" is introduced; it limits the adaptive pseudo-control  $v_a$  and  $v$  by the mean of one component which represents an estimation of the actuator's dynamic (PCH – Pseudo control Hedging). PCH "moves back the reference model" introducing a correction of the reference model's response; this correction depends on actuator's position [3], [9]. Because the dependence between  $\delta$  and  $\delta_c$  is expressed by a non-linear function  $h_a$ , one yields

$$\hat{h}_r(x, \delta_c) \neq \hat{h}_r(x, \hat{\delta}); \quad (52)$$

it results a difference between the two functions

$$v_h = \hat{h}_r(x, \delta_c) - \hat{h}_r(x, \hat{\delta}); \quad (53)$$

Taking into account that

$$\hat{h}_r(x, \delta_c) = \hat{h}_r(x, \hat{h}_r^{-1}(x, v)) = v, \quad (54)$$

function (53) becomes

$$v_h = v - \hat{h}_r(x, \hat{\delta}). \quad (55)$$

This signal is introduced in the reference model as an additional input [3]; one compares it with  $\bar{y}^{(r)}$  inside of the reference model and, after integration, it will lead to the modify of the signals  $\bar{y}$  and  $\hat{y}$ . The block diagram of the subsystem formed by (55) and actuator is presented in fig.5.

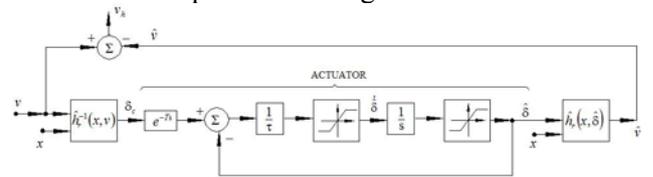


Fig.5. The model of the non-linear actuator

In the case of non-linear actuator for the case of the longitudinal movement of the helicopter (equation (33)), the system from fig.3 includes the model of non-linear actuator (fig.5), in which  $x = \dot{\theta}$ ; the block of calculus for (32) is replaced with the subsystem from fig.5. One choose  $T = 0.03s$  and the control limits in position and speed of the actuators 5grd, respectively 50grd/s [9].

In fig.6 Matlab/Simulink model for the structure from fig.3 is presented; one has chosen  $\theta_c = 5grd$ .

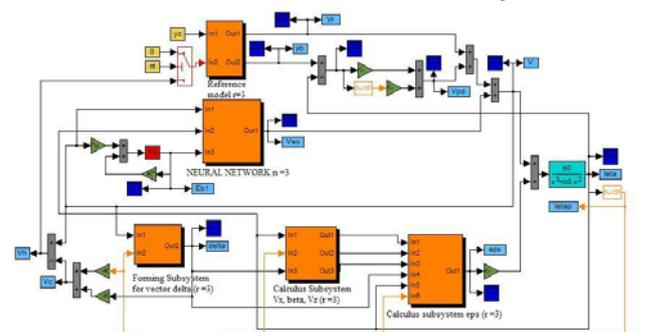


Fig.6. Matlab/Simulink model for the structure from fig.3

In fig.7 the functions  $\bar{\theta}(t), \theta(t), \varepsilon(t), \hat{v}_a(t), \hat{\delta}(t), \delta(t)$  and  $v(t)$  ( $\bar{\theta}, \varepsilon, \hat{\delta}$  – with blue color and  $\theta, \hat{v}_a, \delta$  with red color) are presented. If the actuator is a linear one  $\theta \rightarrow \bar{\theta}, \hat{v}_a \rightarrow \varepsilon$  (the adaptive component of the command compensates  $h_r$  approximation's error),  $\delta \rightarrow \hat{\delta}$  and  $v \rightarrow 0$ .

If the actuator is non-linear one obtains the characteristics from fig.8; additionally, characteris-

tics  $v_h(t)$  and  $\dot{\theta}(\theta)$  appear. When  $v_h = 0$  the actuator is in the saturation state and it works in the linear zone when  $v_h \neq 0$ . The characteristic  $\dot{\theta}(\theta)$  (phase portrait of the system) shows that the non-linear system tends to a stable limit cycle.

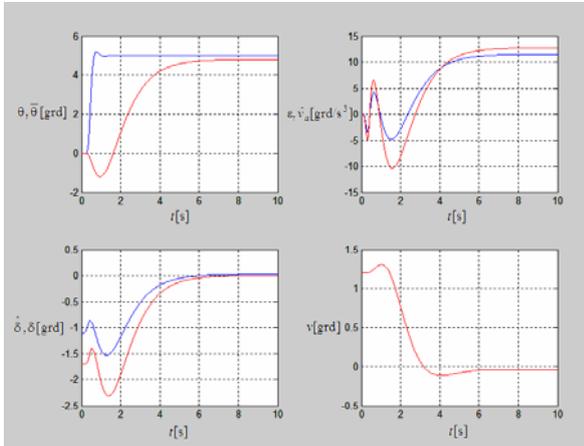


Fig.7. Time characteristics in the case of linear actuator's use ( $r = 3$ )

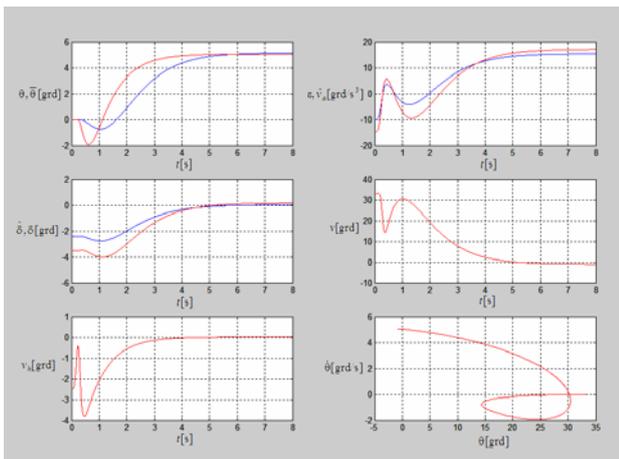


Fig.8. Time characteristics in the case of non-linear actuator's use ( $r = 3$ )

#### 4 Conclusion

The aim of the adaptive command is to compensate the dynamic inversion error. Thus, the command law has two components: the command given by the linear dynamic compensator and the adaptive command given by the neural network. As control system one chooses the non-linear model of helicopter's dynamics in longitudinal plain. The reference model is linear. One obtains the structure of the adaptive control system of the pitch angle and Matlab/Simulink models of the adaptive command system's subsystems. Using these, some characteristics families are obtained; these describe the adaptive command system's dynamics with linear or non-linear actuator.

#### References:

- [1] Calise, A.J. *Flight Evaluation of An Adaptive Velocity Command System for Unmanned Helicopters*. AIAA Guidance, Navigation and Control Conference and Exhibit, 11–14 August, 2003, Austin, Texas.
- [2] Hoseini, S.M., Farrokhi, M., Kashkonei, A.J. Robust Adaptive Control Systems Using Neural Networks. *The International Journal of Control*, 2006, 7 pp.
- [3] Johnson, E.N., Calise, A.J. *Adaptive Guidance and Control for Autonomous Launch Vehicles*. IEEE Aerospace Conference, Big Sky, MT, April, 2001, In Proceedings, 13 pp.
- [4] Lungu, M. *Sisteme de conducere a zborului*. Sitech Publisher, Craiova, 2008, 329 pp. ISBN 978-973-746-975-5.
- [5] Calise, A.J., Hovakymyan, N., Idan, M. *Adaptive Output Control of Nonlinear Systems Using Neural Networks*. Automatica, 37 (8), August, 2001, pp. 1201 – 1211.
- [6] Gregory, L.P. *Adaptive Inverse Control of Plants with Disturbances*. Stanford University, 1998.
- [7] Hovakimyan, N., Nardi, F., Kim, N., Calise, A.J. Adaptive Output Feedback Control of Uncertain Systems Using Single Hidden Layer Neural Networks. *IEEE Transactions on Neural Networks*, 13 (6), 2002, 22 pp.
- [8] Sharma, M., Calise, A.J., *Adaptive Trajectory Control for Autonomous Helicopters*. AIAA Guidance, Navigation and Control Conference, 14 – 17 August, 2001, Montreal, Canada. In Proceedings, 9 pp.
- [9] Hovakimyan, N., Kim, N., Calise, A.J., Parasad, J.V.R. *Adaptive Output Feedback for High – Bandwidth Control of an Unmanned Helicopter*. AIAA Guidance, Navigation and Control Conference, 6 – 9 August, Montreal, Canada, 11 pp.